INTEL ADVISOR:
ROOFLINE AUTOMATION COMMAND LINE AND GUI

Intel Software and Services, 2017

Zakhar Matveev, PhD, Product architect

Roofline access and how-to

- For 2017 Update 1

  (!) Requires env variable set before running command line or GUI:

  ```
  export ADVIXE_EXPERIMENTAL=roofline
  ```

- Starting from 2017 Update 2

  ```
  Just available by default
  ```

Roofline access and how-to **command line** example

FLOP/S = #FLOP/Seconds

```
(optional) > source advixe-vars.sh

(optional) > export ADVIXE_EXPERIMENTAL=roofline
```

1st pass
Obtain "Seconds"
1.1x overhead

```
> advixe-cl --collect survey -no-auto-finalize --project-dir ./your_project
-- <your-executable-with-parameters>
```

2nd pass
Obtain #FLOP count:
3x-5x overhead

```
> advixe-cl --collect tripcounts -flops-and-masks --project-dir
./your_project -- <your-executable-with-parameters>
```

Launch GUI

```
> advixe-gui ./your_project
```

MPI and Cori-specific

1st step:

```
srun -n <num-of-ranks> -c <num_of_cores_per_rank> advixe-cl -v -collect
survey -project-dir=<same_dir_name> -data-limit=0 <your_executable>
```

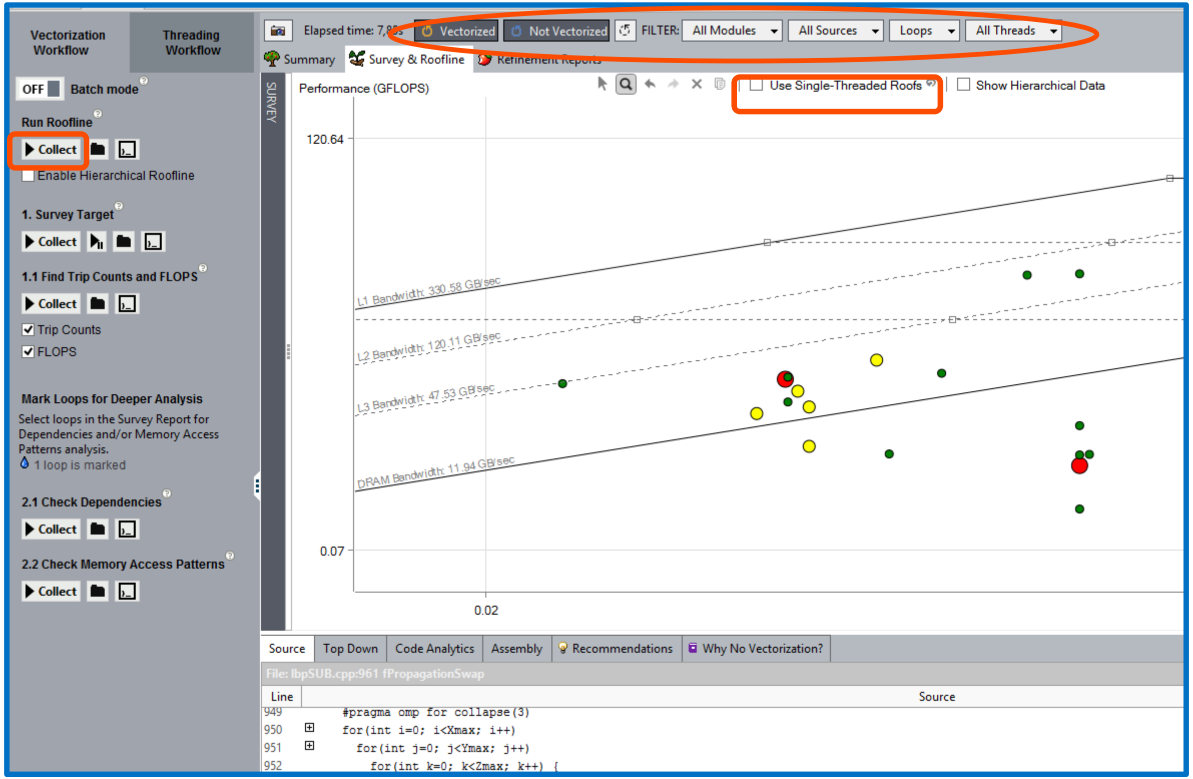
2nd step:

```
srun -n <num-of-ranks> -c <num_of_cores_per_rank> advixe-cl -v -collect
tripcounts -flops-and-masks -project-dir=<same_dir_name> -data-limit=0
<your_executable>
```

Don’t use home dir, use

```
cd $SCRATCH
```

Roofline GUI access and how-to: **GUI**



Select loops in the Survey Report for Dependencies and/or Memory Access Patterns analysis.

1 loop is marked

File: lbpSUB.cpp:961 fPropagationSwap

```
#pragma omp for collapse(3)
for(int i=0; i<Xmax; i++)
  for(int j=0; j<Ymax; j++)
    for(int k=0; k<Zmax; k++) {
```

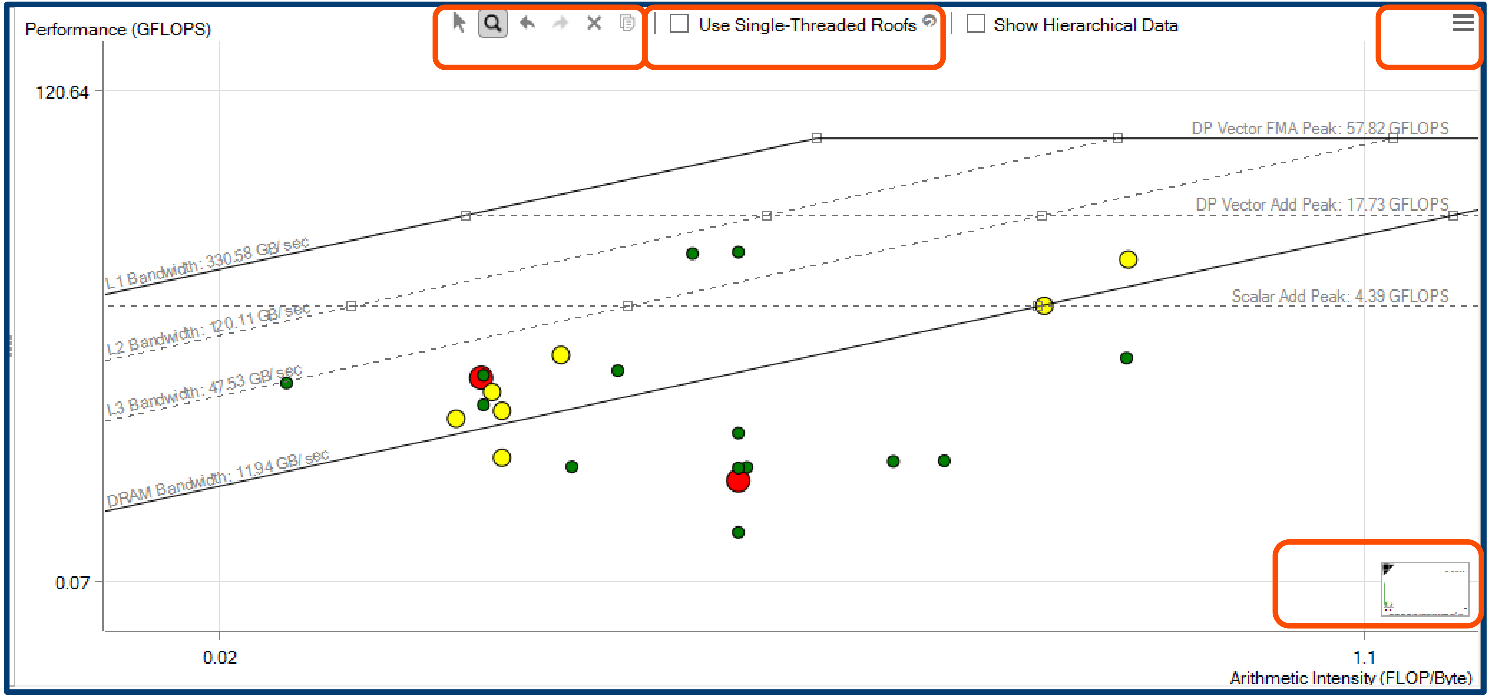
1) <u>“**Run Roofline**” is most automated way.</u>

2) You can also use 2 separate runs:

1. Survey
2. TripCounts (remember to switch **FLOPs** ON)

3) Batch Mode

Roofline Chart



Trouble-shooting

Observe slower Suvey analysis or “finalization”?
(1.5x slower than native run and more)

Change default call stacks processing mode (especially for Fortran)

```
-collect survey –stackwalk-mode=online –no-stack-stitching
```

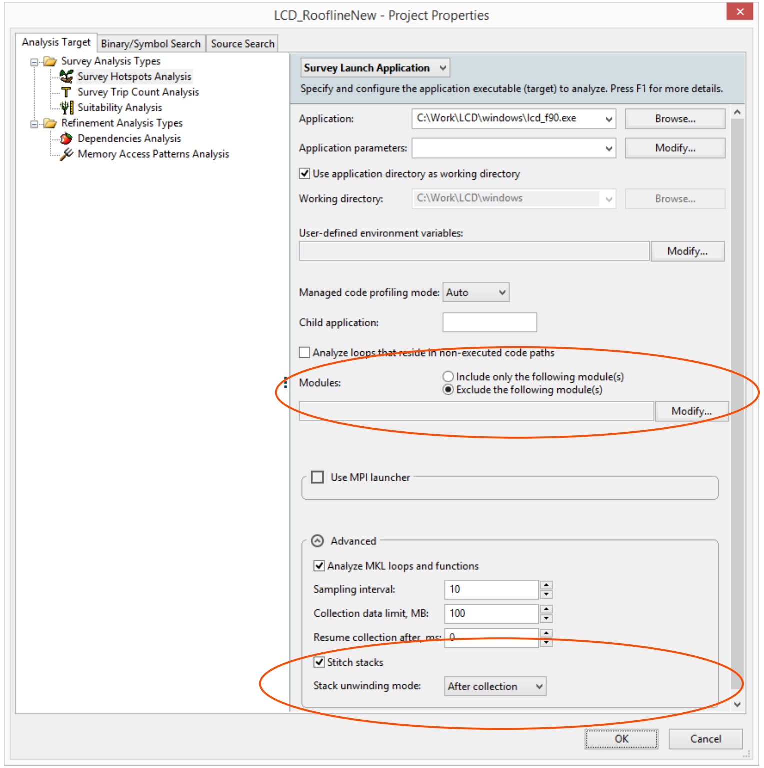
Consider disabling system modules and non-interesting moduels processing:

```
-collect survey -module-filter-mode=include -module-filter=foo.so
```

Observe slower Suvey analysis or “finalization”?

(1.5x slower than native run and more)

Same thing is configurable via GUI:



LCD_RooflineNew - Project Properties

Specify and configure the application executable (target) to analyze. Press F1 for more details.

Application: C:\Work\LCD\windows\lcd_f90.exe

Application parameters:

☑ Use application directory as working directory

Working directory: C:\Work\LCD\windows

User-defined environment variables:

Managed code profiling mode: Auto

Child application:

☐ Analyze loops that reside in non-executed code paths

Modules: ○ Include only the following module(s) ● Exclude the following module(s)

☐ Use MPI launcher

Advanced

☑ Analyze MKL loops and functions

Sampling interval: 10

Collection data limit, MB: 100

Resume collection after, ms: 0

☑ Stitch stacks

Stack unwinding mode: After collection

Observe slow tripcounts/FLOP analysis ??
(> 8x slower than native and more)

FLOPS only , disable TripCounts:

```
-collect tripcounts –flops-and-masks –no-trip-counts
```

TripCounts only, disable FLOPS (No Roofline):

```
-collect tripcounts
```

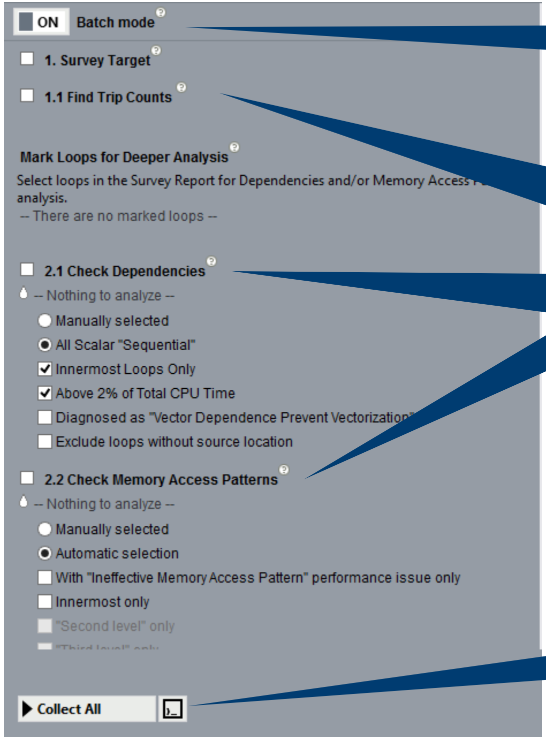
TripCounts and FLOPS:

```
-collect tripcounts –flops-and-masks
```

BACKUP

Batch Mode Workflow Saves Time

Intel® Advisor - Vectorization Advisor



ON Batch mode

□ 1. Survey Target

□ 1.1 Find Trip Counts

Mark Loops for Deeper Analysis

Select loops in the Survey Report for Dependencies and/or Memory Access P
analysis.

-- There are no marked loops --

□ 2.1 Check Dependencies

-- Nothing to analyze --

○ Manually selected

● All Scalar "Sequential"

☑ Innermost Loops Only

☑ Above 2% of Total CPU Time

□ Diagnosed as "Vector Dependence Prevent Vectorization"

□ Exclude loops without source location

□ 2.2 Check Memory Access Patterns

-- Nothing to analyze --

○ Manually selected

● Automatic selection

□ With "Ineffective Memory Access Pattern" performance issue only

□ Innermost only

□ "Second level" only

□ "Third level" only

Turn On Batch Mode

Select analyses to run

Click Collect all

Run several analyses in batch as a single run

Contains pre-selected criteria for advanced analyses

Optimization Notice

Legal Disclaimer & Optimization Notice

INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS". NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO THIS INFORMATION INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

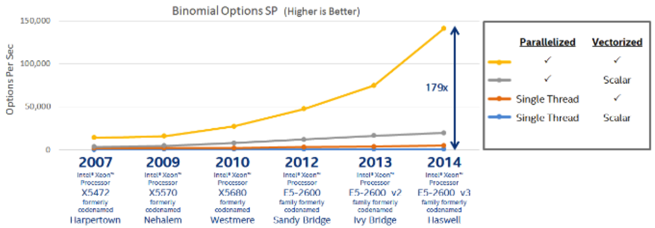
Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

Configurations for Binomial Options SP



Optimization Notice
Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804

Performance measured in Intel Labs by Intel employees

Platform Hardware and Software Configuration

Platform	Unscaled Core Frequency	Cores/ Socket	Num Sockets	L1 Data Cache	L1 I Cache	L2 Cache	L3 Cache	Memory	Memory Frequency	Memory Access	H/W Prefetchers Enabled	HT Enabled	Turbo Enabled	C States	O/S Name	Operating System	Compiler Version
Intel® Xeon™ 5472 Processor	3.0 GHZ	4	2	32K	32K	12 MB	None	32 GB	800 MHZ	UMA	Y	N	N	Disabled	Fedora 20	3.11.10-301.fc20	icc version 14.0.1
Intel® Xeon™ X5570 Processor	2.93 GHZ	4	2	32K	32K	256K	8 MB	48 GB	1333 MHZ	NUMA	Y	Y	Y	Disabled	Fedora 20	3.11.10-301.fc20	icc version 14.0.1
Intel® Xeon™ X5680 Processor	3.33 GHZ	6	2	32K	32K	256K	12 MB	48 MB	1333 MHZ	NUMA	Y	Y	Y	Disabled	Fedora 20	3.11.10-301.fc20	icc version 14.0.1
Intel® Xeon™ E5 2690 Processor	2.9 GHZ	8	2	32K	32K	256K	20 MB	64 GB	1600 MHZ	NUMA	Y	Y	Y	Disabled	Fedora 20	3.11.10-301.fc20	icc version 14.0.1
Intel® Xeon™ E5 2697v2 Processor	2.7 GHZ	12	2	32K	32K	256K	30 MB	64 GB	1867 MHZ	NUMA	Y	Y	Y	Disabled	Fedora 20	3.11.10-301.fc20	icc version 14.0.1
Codename Haswell	2.2 GHz	14	2	32K	32K	256K	35 MB	64 GB	2133 MHZ	NUMA	Y	Y	Y	Disabled	Fedora 20	3.13.5-202.fc20	icc version 14.0.1

intel®

experience
what's inside™